

BAB II

LANDASAN TEORI

1.1 Tinjauan Pustaka

Edy Bahraen (2018), telah membuat sebuah aplikasi pemesanan lapangan di Pelle Futsal. Dalam penelitiannya, penulis mengimplementasikan web *responsive* yang menggunakan fitur ekstensi *bootstrap* dan *framework* *YII* agar dapat mempermudah proses pemesanan lapangan Pelle Futsal serta desain yang *responsive* dapat membantu dalam *user interface* atau tampilan berdasarkan layar *device* seperti *mobile (handphone)*, Ipad, tablet, laptop dan lain sebagainya.

Ester Puspitanungrum (2015), telah membuat aplikasi pemesanan Koran di PT.SUARA MERDEKA, Semarang. Dalam penelitiannya, penulis menerapkan *framework* *YII* untuk membangun aplikasi berbasis web, untuk pemesanan koran, dan pengelolaan data pesanan koran untuk menciptakan sistem yang *real time*.

Mursito (2017), telah membuat aplikasi TAS (Tugas Akhir Skripsi) di Fakultas MIPA Universitas Negeri Yogyakarta. Dalam penelitiannya, penulis mengimplementasikan fitur *Firebase Cloud Messaging* (FCM) yang merupakan *mobile backend as a service* (MBaaS) yang dimiliki oleh Google. Penelitian tersebut ditujukan untuk memfasilitasi proses bimbingan TAS sebagai alternatif kedua agar memudahkan mahasiswa dan dosen dalam melakukan bimbingan, kapan dan dimana saja secara online sesuai dengan persyaratan yang telah ditentukan.

Riduansyah (2017), telah membuat aplikasi media *broadcast* informasi umum dan jadwal kuliah di STMIK Akakom Yogyakarta. Dalam penelitiannya, penulis mengimplementasikan fitur *Google Cloud Messaging* (GCM) yang akan mengirimkan sebuah *broadcast* notifikasi berupa informasi umum dan jadwal kuliah ke seluruh pengguna aplikasi (mahasiswa). Pada aplikasi ini juga menggunakan sebuah *system server* berbasis *desktop application* untuk menyebarkan informasi secara *realtime* dan sistem *client* berbasis android yang digunakan sebagai media penerima informasi.

Perbandingan penelitian yang akan diajukan oleh Lina Dwi Handayani (2018) adalah penerapan *framework* YII versi 2 untuk membuat sebuah aplikasi pemesanan kerajinan untuk mempermudah agen serta admin dalam mengorganisir data pesanan. Aplikasi yang dibuat akan terintegrasi dengan perangkat android yang diharapkan agar dapat mempermudah admin dalam memberitahukan pesanan dengan mengimplementasikan fitur teknologi *firebase cloud messaging* (FCM).

Adapun perbandingan tinjauan pustaka tersebut dapat di lihat pada Tabel 2.1:

Tabel 1.1 Tinjauan Pustaka

Sumber	Teknologi	Objek	Masalah	Hasil
Edy Bahraen (2018)	Yii dan ekstensi bootstrap	Lapangan palle futsal	Pemesanan lapangan masih menggunakan sistem manual dan pemesan tidak dapat melihat jadwal waktu lapangan yang sudah dipesan	Sistem pemesanan online berbasis web <i>responsive</i>

Ester Puspitaningrum (2015)	Yii	Koran di PT.SUARA MERDEKA, Semarang	Sistem pemesanan dan pendistribusian koran dan pembuatan laporan penjualan yang masih manual.	Yii dalam perancangan sistem pemesanan dan distribusi koran seperti data kelola user agen, pemesanan dan laporan.
Musrito (2017)	Firebase Cloud Messagin g	Logbook TAS	Seringkali mahasiswa lupa dengan jadwal bimbingan akhir skripsi.	Aplikasi yang dapat menampilkan notifikasi bimbingan kepada mahasiswa yang akan melakukan TAS (Tugas Akhir Skripsi)
Riduansyah (2017)	Google Cloud Messagin g	Informasi Umum Jadwal Kuliah di STMIK Akakom Yogyakarta	Sebelumnya pemberitahuan informasi jadwal kuliah masih disebarkan secara manual.	Notifikasi pembaharuan informasi di STMIK Akakom Yogyakarta
Usulan Lina Dwi Handayani	Yii 2 dan Firebase Cloud Messagin g	Kerajinan Tenun (Studi Kasus Lawe Home Craft Industry)	Pemesanan masih manual dimana agen harus datang ke showroom atau melalui telephone dan pengiriman data pesanan kepada agen hanya melalui sms atau datang ke rumah bagian produksi.	Aplikasi pemesanan berbasis web dengan menggunakan framework Yii 2 dalam perancangan sistem frontend dan backend nya serta Firebase Cloud Messaging untuk mengirim sebuah push notifikasi.

1.2 Dasar Teori

1.2.1 Lawe Home Industry

Lawe adalah perusahaan sosial masyarakat yang mengubah kerajinan tangan tradisional menjadi produk fungsional melalui pemberdayaan perempuan. Peluang bisnis menawarkan bagi perempuan Indonesia untuk mengembangkan

potensi mereka. Ide untuk menciptakan produk lebih terfokus pada 'apa yang dapat membuat para pengrajin bekerja', tidak semata-mata mendukung kebutuhan pasar, proses menjembatani antara pengrajin ke pasar.

Lawe memproduksi kerajinan tangan dengan tenunan tradisional sebagai bahan dasarnya. Penyediaan bahan dasar dilakukan melalui kolaborasi dengan para penenun, dalam sistem langsung. Tenunan tersebut kemudian diproses selama produksi menjadi produk siap pakai. Desain produk dilakukan dengan menyesuaikan kapasitas dan kemampuan pengrajin. Para perajin yang sebagian besar ibu rumah tangga akan membawa produksi di rumah mereka sendiri. Lawe akan mengumpulkan produk setelah selesai dan melakukan kontrol kualitas. Jika produk lolos, maka akan siap untuk dipasarkan.

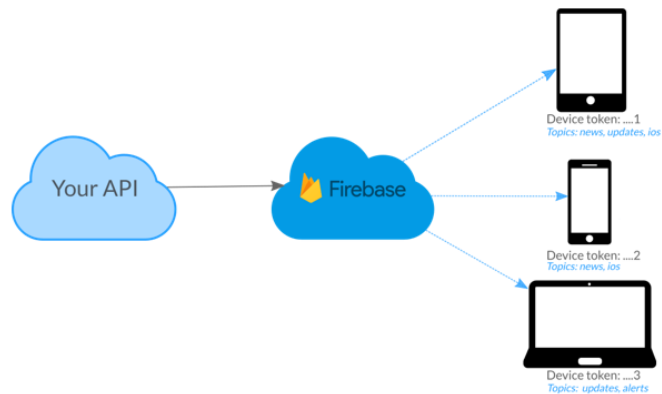
1.2.2 Lurik (Ensiklopedi Nasional Indonesia, 1997)

Lurik menurut Ensiklopedi Nasional Indonesia (1997) adalah suatu kain hasil tenunan benang yang berasal dari daerah Jawa Tengah dengan motif dasar garis-garis atau kotak-kotak dengan warna-warna suram yang pada umumnya diselingi aneka warna benang. Kata lurik berasal dari akar kata rik yang artinya garis atau parit yang dimaknai sebagai pagar atau pelindung bagi pemakainya. Menurut Kamus Besar Bahasa Indonesia (1990), lurik adalah kain tenun yang memiliki corak jalur-jalur, sedangkan dalam Kamus Lengkap Bahasa Jawa (Mangunsuwito:20 02) pengertian lurik adalah corak lirik-lirik atau lorek-lorek, yang berarti garis-garis dalam bahasa Indonesia.

Dan berbagai definisi yang telah disebutkan di atas, dengan demikian dapat disimpulkan bahwa lurik merupakan kain yang diperoleh melalui proses penenunan dari seutas benang (lawe) yang diolah sedemikian rupa menjadi selembur kain katun. Proses yang dimaksud yaitu diawali dari pembuatan benang tukel, tahap pencelupan yaitu pencucian dan pewarnaan, pengelosan dan pemaletan, penghanian, pencucuk-an, penyetelan, dan penenunan. Motif atau corak yang dihasilkan berupa garis-garis vertikal maupun horisontal yang dijalin sedemikian rupa sesuai warna yang dikehendaki dengan berbagai variasinya.

1.2.3 *Firebase Cloud Messaging*

Notifikasi akan memudahkan pengguna untuk mendapatkan informasi baik informasi pengingat ataupun informasi lainnya. Beberapa teknologi yang menyediakan *tools* notifikasi salah satunya adalah *Firebase Cloud Messaging* (FCM). *Firebase* adalah sebuah penyedia layanan berupa *database realtime* dan *backend* yang digunakan pada berbagai *platform*. *Backend* sendiri adalah sebuah bagian dalam kode aplikasi yang berhubungan langsung dengan isi *database*. Dengan *firebase*, pengembang aplikasi tidak perlu membuat *backend* sendiri melainkan memakai API yang telah disediakan oleh *firebase* sehingga pengembangan aplikasi dapat dipersingkat. *Firebase* dikembangkan dengan menggunakan database MongoDB sehingga *firebase* menggunakan tipe database NoSQL. Karena memakai tipe database NoSQL maka struktur database dari *firebase* bersifat fleksibel dan cepat sehingga cocok untuk digunakan pada aplikasi berbasis *mobile*.



Gambar 1.1 Arsitektur *Firebase Cloud Messaging*

1.2.4 *Push Notification* (Alexander Laysha ,2015)

Push notification adalah pesan pendek yang muncul secara otomatis di layar smartphone. Implementasi dari salah satu fitur yang dimiliki oleh *Firebase Cloud Messaging* ini mencakup 2 komponen utama untuk mengirim dan menerima pesan, komponen yang pertama adalah lingkungan tepercaya seperti *Cloud Functions for Firebase* atau server aplikasi yang akan digunakan untuk membuat, menargetkan, dan mengirim pesan. Komponen yang kedua adalah aplikasi *client* *iOS*, *Android*, atau *Web* (*JavaScript*) yang menerima pesan. Tidak seperti skenario tradisional (*pull*) dimana klien harus *merequest* setiap kali ingin mendapatkan informasi dari sistem. Secara umum, *push notification* dikirimkan melalui *Push Notification Service* (PNS) yang spesifik untuk setiap platform: seperti *Apple Push Notification Service* (APNs) untuk Apple, *Google Cloud Message* (GCM) untuk *Android*, dan *Microsoft Push Notification Service* (MPNS) untuk *Windows Phone*. Namun algoritma untuk tiap PNS sama.



Gambar 1.2 Skema *Push Notification*

Algoritma *push notification* :

1. *Device* android terdaftar dalam PNS (*Push Notification Service*) yaitu dengan cara mengirimkan ID aplikasi (nama paket) ke server PNS.
2. Apabila registrasi berhasil, server PNS akan menerbitkan ID registrasi yaitu berupa token kepada *device* android.
3. Setelah mendapatkan ID registrasi, *device* akan mengirimkan ID registrasi tersebut ke server.
4. Admin mengirimkan *push notification* melalui *backend* ke *device*, dan server akan mengirimkan notifikasi tersebut ke server PNS.
5. PNS meneruskan notifikasi ke *device*.

1.2.5 Yii Framework (Badiyanto, 2016)

Yii adalah *framework* (kerangka kerja) PHP berbasis komponen, dan kinerja terbaik untuk pengembangan aplikasi web berskala besar. Yii menyediakan *reusability* maksimum dalam pemrograman web dan mampu meningkatkan kecepatan pengembangan secara signifikan. Nama Yii diambil dari singkatan “*Yes It Is*”. Yii juga merupakan *framework* yang merespon paling tepat dan akurat. Sehubungan dengan itu, penggunaan *framework* membutuhkan pemahaman dasar – dasar pemrograman secara *procedural* maupun pemrograman berorientasi objek. *Framework* Yii menggunakan dua konsep yang paling penting, yaitu *Object-Oriented Programming* (OOP) dan *Model-View-Controller* (MVC).

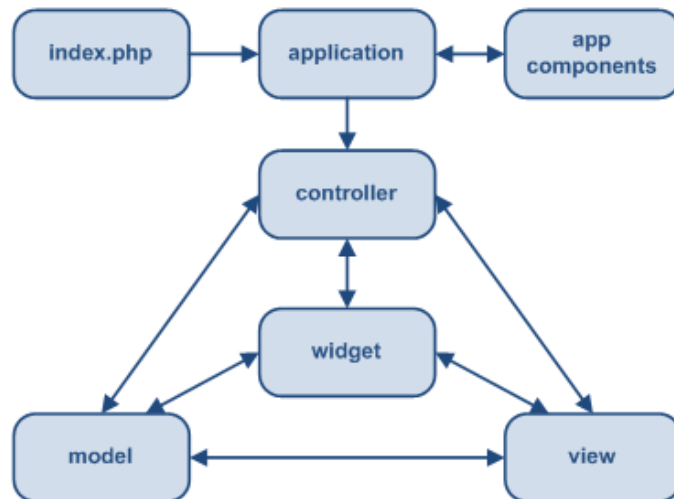
Model-View-Controller (MVC) adalah model pembuatan program yang menerapkan arsitektur yang memisahkan proses, tampilan dan bagian yang menghubungkan antara proses dan tampilan. MVC bertujuan untuk memisahkan proses bisnis dari pertimbangan antarmuka user agar para pengembang bisa lebih mudah mengubah setiap bagian tanpa mempengaruhi yang lain. Sedangkan dalam pola MVC, komponen aplikasi dibagi menjadi tiga bagian, yaitu :

- 1) *Model*, merupakan kelas untuk mendefinisikan aturan-aturan yang digunakan dalam pembuatan aplikasi, seperti fungsi validasi, relasi antar tabel, mendefinisikan tabel, atribut dalam tabel, fungsi pencarian, dan lain-lain.
- 2) *View*, merupakan kelas untuk pengaturan *user interface* atau tampilan dari aplikasi yang dibuat. Di dalam *view* terdapat folder *layout* yang berfungsi untuk mengatur bentuk tampilan dari aplikasi yang akan dibuat. Untuk

memanggil kelas dalam *view*, dapat dipanggil dengan fungsi *render*. Untuk merender *view*, *controller* menjalankan method *CController::render* dengan nama class pada file *view*, kemudian method akan mencari kelas pada file *view* yang terkait

- 3) *Controller*, merupakan kelas penghubung antara model, dan *view*. Biasanya *controller* berisi logika program untuk aplikasi yang dibuat.

Dalam mengimplementasikan pola desain MVC, Yii memperkenalkan bagian *controller* depan (*front-controller*) yang disebut *application*. Elemen ini bertugas untuk memproses permintaan (*request*) yang dikirim oleh *user* dan mengirimnya ke *controller* yang sesuai, untuk proses penanganan lebih lanjut. Dalam beberapa *framework* web lain, elemen ini sering juga disebut sebagai *router*. Berikut ini gambar 2.3 yang menunjukkan arsitektur aplikasi di dalam Yii.



Gambar 1.3 Struktur Aplikasi Yii

